Claims 1-10 (Cancelled.)

11. (Currently Amended) A computerized method for identifying malicious code in a target program running in a virtual machine of a computer system, the method comprising:

evaluating a file format of the target program;

evaluating control fields within a header of a file containing the target program;

automatically configuring a memory map of the virtual machine by assigning areas of the memory map to receive predetermined types of data from the target program based on the file format in order to execute the target program, the virtual machine being capable of executing the target program in one of three modes of operation based on the file format and the control fields within the header of the file, a first mode of operation comprising a real mode for executing programs comprising instructions based on DOS, a second mode of operation for executing target programs comprising a high level programming language, and a third mode of operation comprising a protected mode for executing target programs comprising thirty-two bit code;

constructing the virtual machine from one or more layered operating system shells that correspond with the memory map so that the virtual machine is capable of executing DOS and Windows type target programs;

simulating values of the computer system with the one or more layered operating system shells of the virtual machine;

~~storing~~ setting and resetting behavior flags ~~representing~~ in a register in order to track behavior of the target program in response to the simulated values during execution of the target program by the virtual machine;

storing a sequence in which the behavior flags are set and reset in the register by the target program during execution of the target program by the virtual machine;

passing behavior flag data and sequence flag data from the virtual machine to the computer system for evaluation after execution of the target program by the virtual machine; [[and]]

terminating the virtual machine after execution of the target program, thereby removing from the computer system a copy of the target program that was contained within the virtual machine; and

evaluating the behavior flag data and sequence flag data with the computer system to determine if the target program contains malicious code.

12. (Cancelled).

13. (Previously Presented) The method of Claim 11, further comprising initializing the virtual machine within the computer system, the virtual machine comprising a virtual computer implemented by software simulating functionality of a central processing unit and memory and a virtual operating system simulating functionality of an operating system of the computer system.

14. (Previously Presented) The method of Claim 11, further comprising identifying a type of operating system intended for the target program that is to be executed by the virtual machine.

15-16 (Cancelled).

17. (Previously Presented) The method of Claim 11, wherein the virtual machine executes the target program starting at each entry point defined within an entry point table.

18. (Previously Presented) The method of Claim 11, further comprising loading a software CPU shell when the virtual machine operates in the first and third modes of operation.

19. (Previously Presented) The method of Claim 11, further comprising loading a language interpreter when the virtual machine operates in the second mode of operation.

[The remainder of this page has been intentionally left blank.]

20. (Currently Amended) A computer system for discovering malicious code in a target program, comprising:

a processing unit;

a memory storage device; and

one or more program modules stored in said memory storage device for providing instructions to said processing unit;

said processing unit executing said instructions of said one or more program modules, operable for

evaluating a file format of the target program;

evaluating control fields within a header of a file containing the target program;

automatically configuring a memory map of a virtual machine by assigning areas of the memory map to receive predetermined types of data from the target program based on the file format in order to execute the target program, the virtual machine being capable of executing the target program in one of three modes of operation based on the file format and the control fields within the header of the file, a first mode of operation comprising a real mode for executing programs comprising instructions based on DOS, a second mode of operation for executing target programs comprising a high level programming language, and third mode of operation for executing target programs comprising thirty-two bit code;

constructing the virtual machine from one or more layered operating system shells that correspond with the memory map so that the virtual machine is capable of executing DOS and Windows type target programs;

simulating values of the computer system with the one or more layered operating system shells of the virtual machine;

~~storing~~ setting and resetting behavior flags ~~representing~~ in a register in order to track behavior of the target program in response to the simulated values during execution of the target program by the virtual machine;

storing a sequence in which the behavior flags are set and reset in the register by the target program during execution of the target program by the virtual machine;

passing behavior flag data and sequence flag data from the virtual machine to the computer system after execution of the target program by the virtual machine; and

evaluating the behavior flag data and sequence flag data with the computer system to determine if the target program contains malicious code.

21. (Previously Presented) The system of Claim 20, wherein the processing unit is further operable for terminating the virtual machine after execution of the target program, thereby removing from the computer system a copy of the target program that was contained within the virtual machine.

22. (Previously Presented) The system of Claim 20, wherein the virtual machine comprises a virtual computer implemented by the one or more programs simulating functionality of a central processing unit and memory and a virtual operating system simulating functionality of an operating system of the computer system.

23. (Previously Presented) The system of Claim 20, wherein the processing unit is further operable for identifying a type of operating system intended for the target program that is to be executed by the virtual machine.

24-25 (Cancelled).

26. (Previously Presented) The system of Claim 20, wherein the virtual machine executes the target program starting at each entry point defined within an entry point table.

27. (Currently Amended) The system of Claim 20, wherein the processing unit is further operable for loading a software CPU shell when the virtual machine operates in the first and ~~second~~ third modes of operation.

28. (Previously Presented) The system of Claim 20, wherein the processing unit is further operable for loading a language interpreter when the virtual machine operates in the second mode of operation.

[The remainder of this page has been intentionally left blank.]

29. (Currently Amended) A computer-implemented method for identifying malicious code in a target program comprising:

automatically configuring a memory map of a virtual machine by assigning areas of the memory map to receive predetermined types of data from the target program based on the file format to execute the target program, the virtual machine being capable of executing the target program in one of three modes of operation, a first mode of operation comprising a real mode for executing programs comprising instructions based on DOS, a second mode of operation for executing a target program comprising a high level programming language, and a third mode of operation comprising a protected mode for executing a target program comprising thirty-two bit code;

constructing the virtual machine from one or more layered operating system shells that correspond with the memory map so that the virtual machine is capable of executing DOS and Windows type target programs;

simulating values of the computer system with the one or more layered operating system shells of the virtual machine;

~~storing~~ setting and resetting behavior flags ~~representing~~ in a register in order to track behavior of the target program in response to the simulated values during execution of the target program by the virtual machine;

storing a sequence in which the behavior flags are set and reset in the register by the target program during execution of the target program by the virtual machine;

passing behavior flag data and sequence flag data from the virtual machine to a computer system after execution of the target program by the virtual machine; [[and]]

terminating the virtual machine after execution of the target program, thereby removing from the computer system a copy of the target program that was contained within the virtual machine; and

evaluating the behavior flag data and sequence flag data with the computer system to determine if the target program contains malicious code.

30. (Previously Presented) The computer-implemented method of Claim 29, further comprising evaluating a file format of the target program.